Escuela Colombiana De Ingeniería
Julio Garavito

Seguridad y privacidad TI

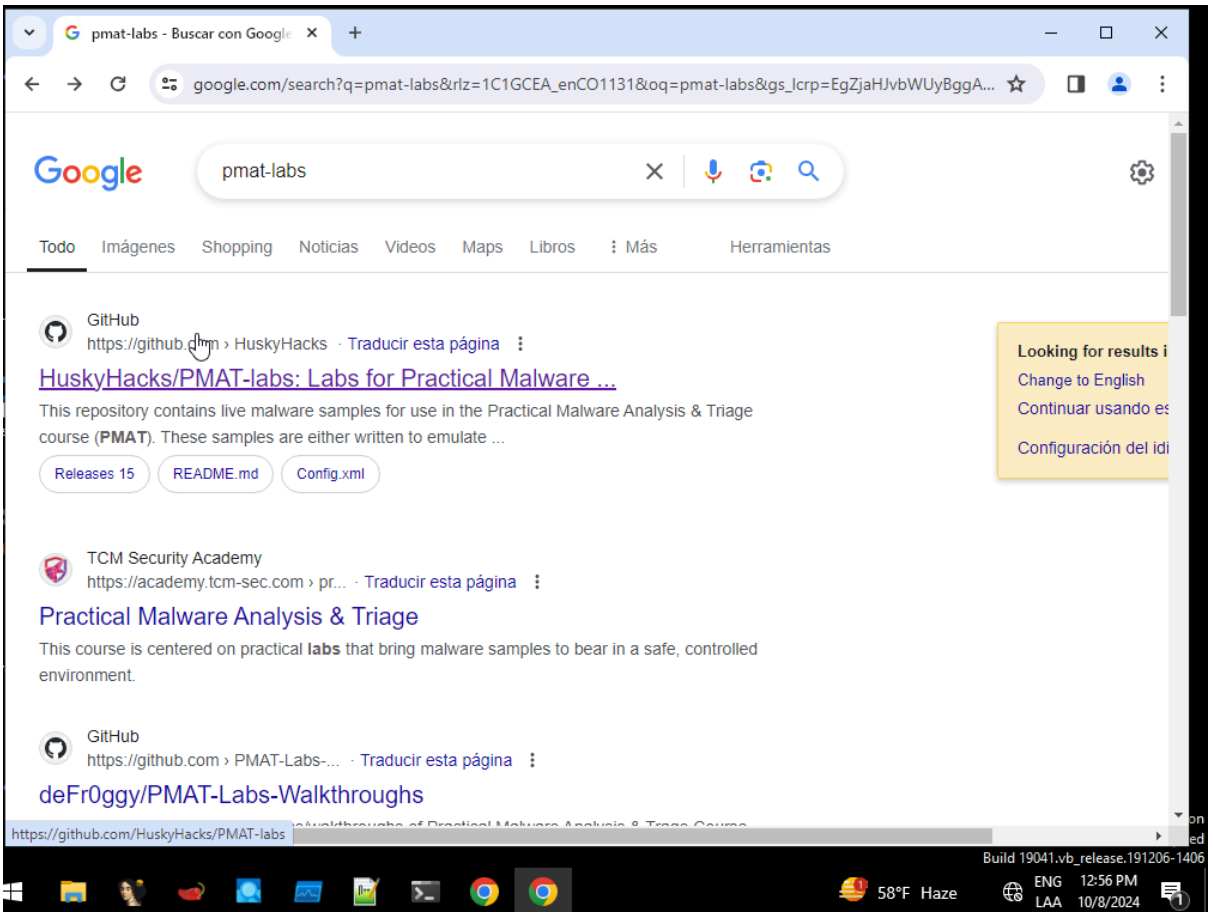
Daniel Esteban Vela Lopez

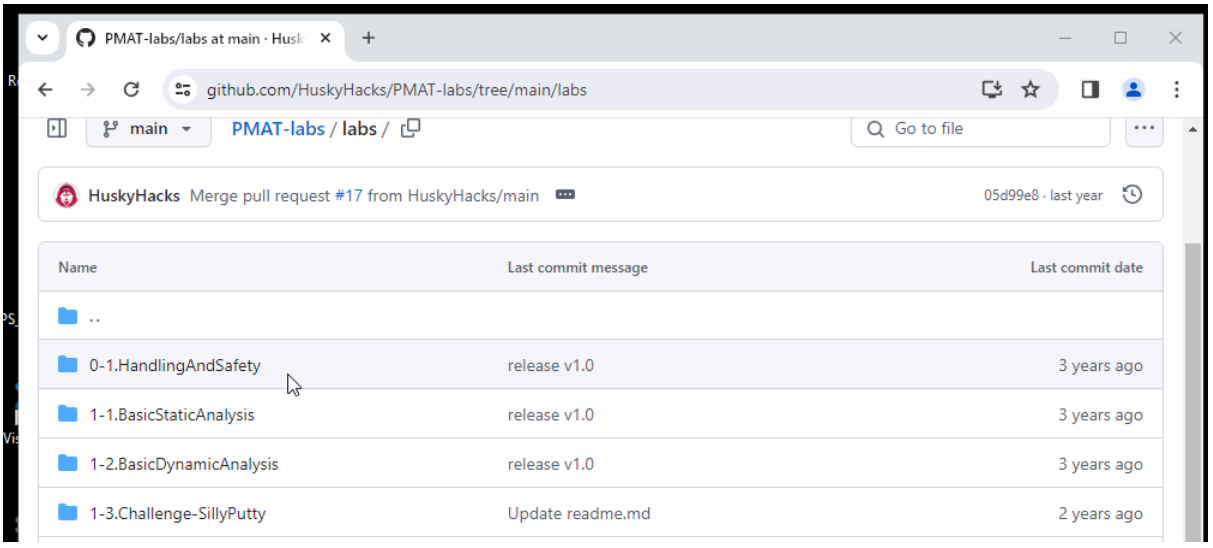
Andrés Felipe Montes

Laura Valentina Rodríguez Ortegón

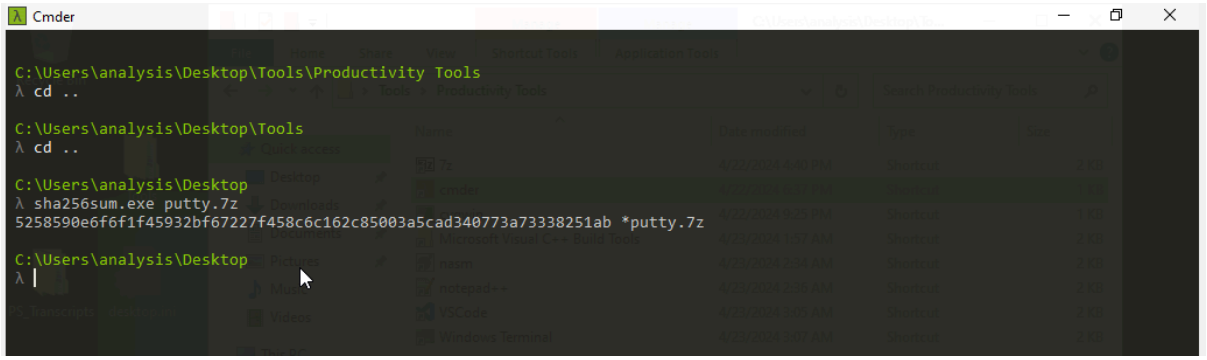
Laboratorio No.8

2024-2





¿Cuál es el hash SHA256 de la muestra?



¿Qué arquitectura es este binario?

PE32 (Portable Executable 32-bit):

- El formato PE (Portable Executable) es el formato estándar para ejecutables, bibliotecas (.dll), y otros archivos de objetos en sistemas operativos Windows. El número 32 indica que el binario es de 32 bits, lo que implica que está diseñado para ejecutarse en arquitecturas de 32 bits.

- Los archivos PE contienen metadatos, secciones de código, recursos y otros elementos que el sistema operativo necesita para cargar y ejecutar el programa.

GUI (Graphical User Interface):

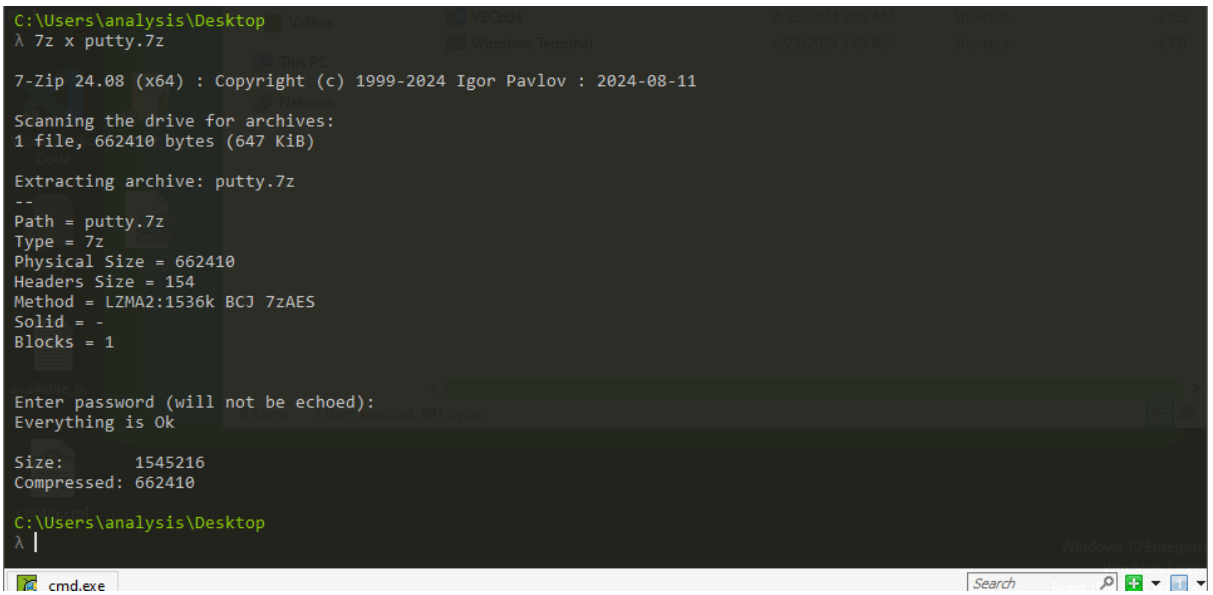
- Indica que el binario es una aplicación que proporciona una interfaz gráfica de usuario. Esto significa que el programa no es puramente de consola (CLI), sino que utiliza ventanas, botones, menús, etc., como se esperaría de una aplicación normal de Windows.

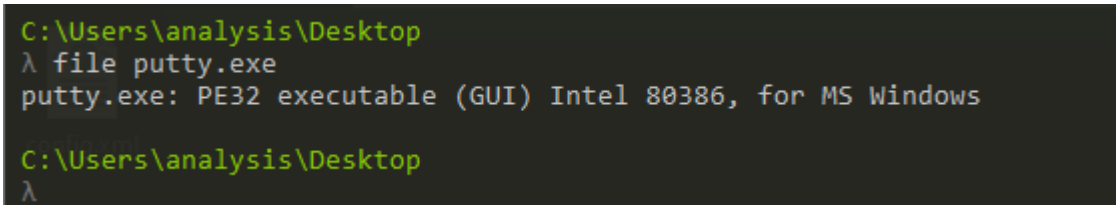
Intel 80386 (i386):

- El hecho de que el binario esté compilado para i386 significa que está destinado a ejecutarse en cualquier CPU compatible con las instrucciones x86 de 32 bits, lo cual es común en entornos Windows más antiguos o en aplicaciones que no necesitan acceso a un gran espacio de memoria.

Para MS Windows:

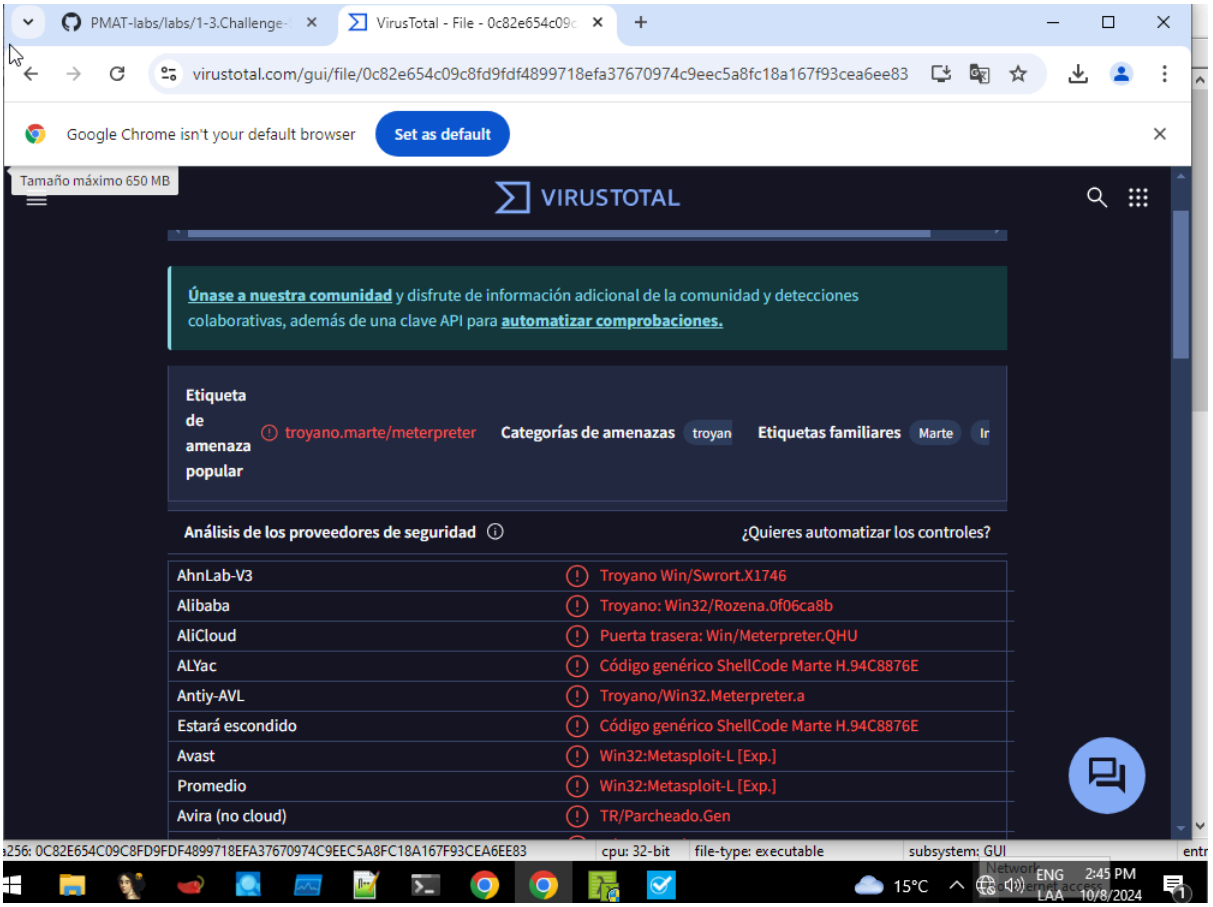
- Indica que este ejecutable está diseñado específicamente para ejecutarse en el sistema operativo Microsoft Windows.





¿Hay algún resultado al enviar el hash SHA256 a VirusTotal?

Al cargar el ejecutable en VirusTotal, notamos algo preocupante: más de 20 proveedores de seguridad han marcado este archivo como un troyano. Este nivel de acuerdo entre distintas plataformas de seguridad es una clara señal de alerta. Cada uno de estos servicios utiliza distintas bases de datos para detectar comportamientos maliciosos, lo que refuerza la validez del análisis. El hecho de que tantas soluciones coincidan en clasificar este archivo como una amenaza subraya el riesgo potencial que representa.



Describe los resultados de mover los hilos de este binario. Registrar y describir cualquier cadena que sea potencialmente interesante. ¿Puede haber alguna información interesante?

FLOSS STACK STRINGS (8):
Esta sección muestra cadenas que se encontraron en la pila (stack) del programa cuando se analiza su flujo de ejecución. Las cadenas pueden incluir:

- Proxy error: Esto parece ser un mensaje de error relacionado con problemas de proxy, indicando que el binario podría estar haciendo uso de una conexión proxy o manejando conexiones de red.

- 0WB4, 1WB4, etc.: Estas cadenas parecen ser identificadores que FLOSS extrajo de la pila. No es claro qué representan, pero podrían ser datos temporales que se almacenan durante la ejecución.

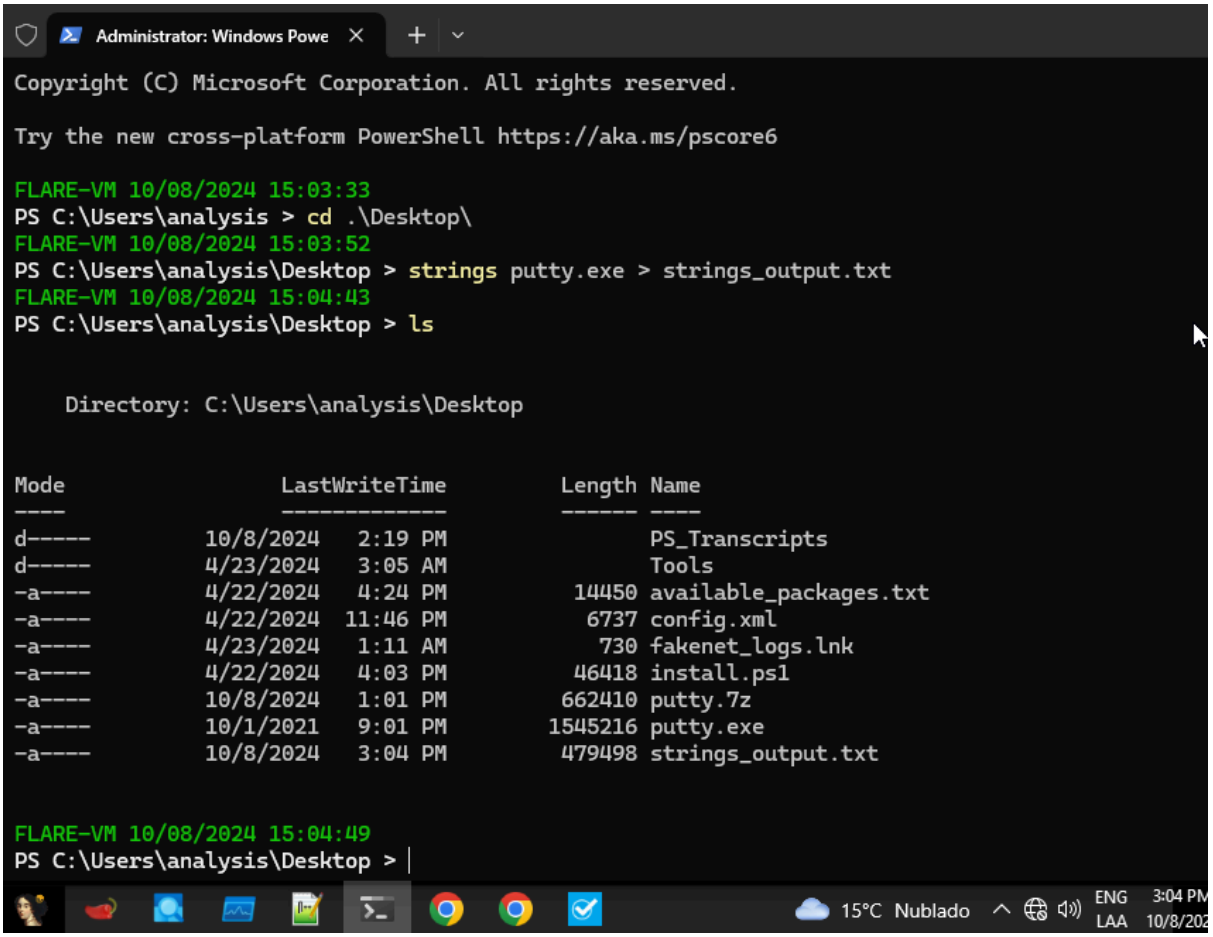
- 0.0.0.0: Esta dirección IP se utiliza comúnmente en programación de red para referirse a todas las interfaces de red locales. Esto podría sugerir que el binario maneja conexiones de red.

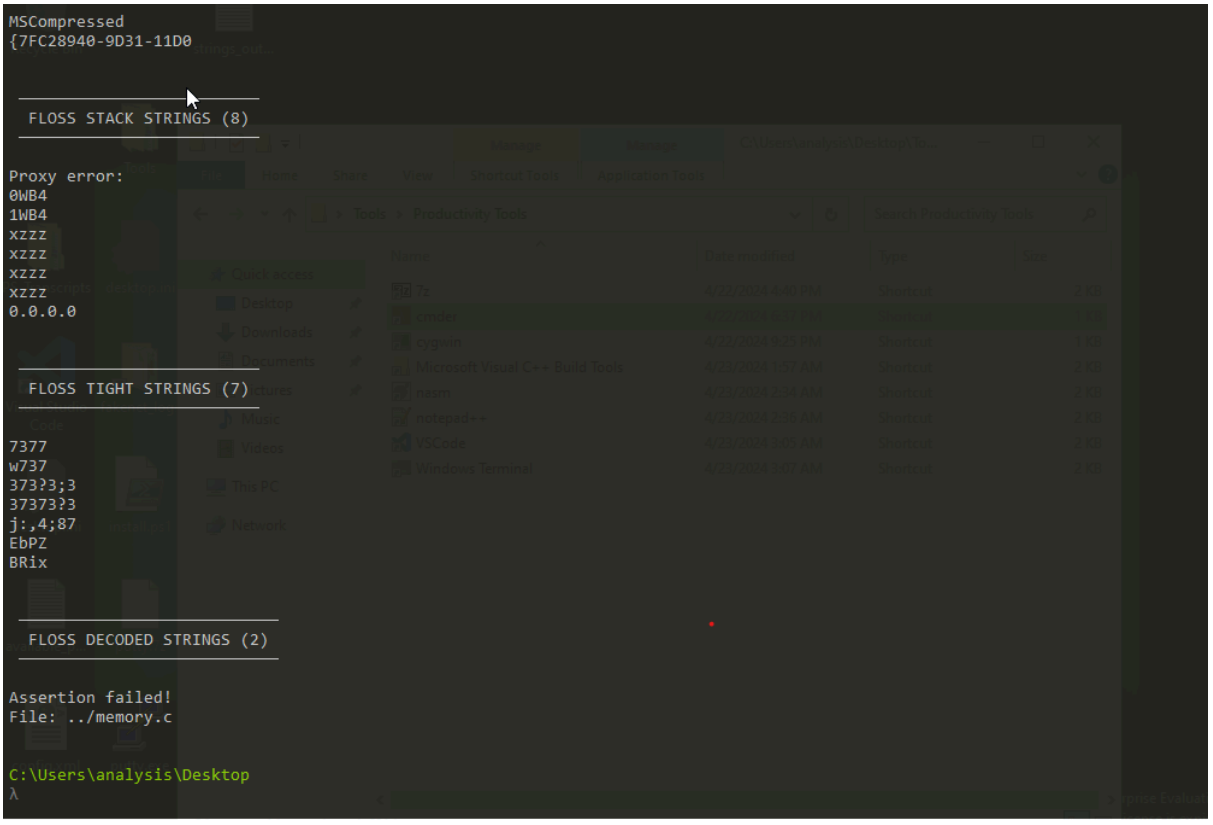
- BRiX: Este es un término interesante, podría ser una palabra clave, parte de un nombre de función, o incluso un nombre de usuario o identificador dentro del programa.

FLOSS DECODED STRINGS (2):
Estas son cadenas que FLOSS ha logrado decodificar. A menudo, los binarios ofuscan o codifican ciertos textos, y FLOSS intenta revelar esas cadenas. Aquí hay dos cadenas decodificadas:

- Assertion failed!: Esta cadena es un mensaje de error común en programación que indica que una condición esperada no se cumplió. Sugiere que el binario tiene validaciones internas y, si algo sale mal, este error podría ser mostrado.

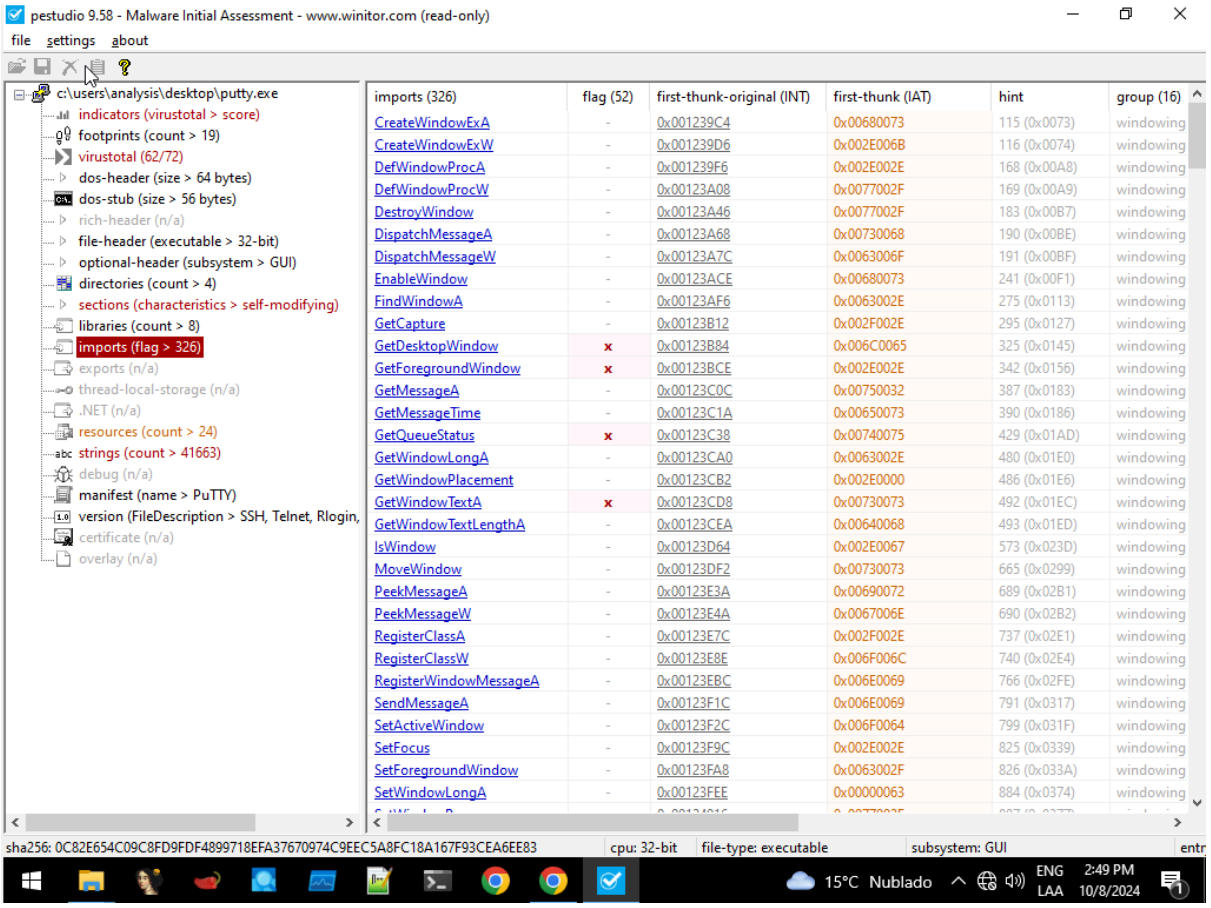
- File: ../memory.c: Esto sugiere que el programa fue escrito en C o C++, y hace referencia a un archivo de código fuente llamado memory.c. Es posible que haya una operación de memoria que podría fallar si no se cumplen ciertas condiciones, lo que causaría el mensaje de error anterior.


Administrator: Windows Powe
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
FLARE-VM 10/08/2024 15:03:33
PS C:\Users\analysis > cd .\Desktop\
FLARE-VM 10/08/2024 15:03:52
PS C:\Users\analysis\Desktop > strings putty.exe > strings_output.txt
FLARE-VM 10/08/2024 15:04:43
PS C:\Users\analysis\Desktop > ls
Directory: C:\Users\analysis\Desktop
Mode LastWriteTime Length Name
d----- 10/8/2024 2:19 PM PS_Transcripts
d----- 4/23/2024 3:05 AM Tools
-a---- 4/22/2024 4:24 PM 14450 available_packages.txt
-a---- 4/22/2024 11:46 PM 6737 config.xml
-a---- 4/23/2024 1:11 AM 730 fakenet_logs.lnk
-a---- 4/22/2024 4:03 PM 46418 install.ps1
-a---- 10/8/2024 1:01 PM 662410 putty.7z
-a---- 10/1/2021 9:01 PM 1545216 putty.exe
-a---- 10/8/2024 3:04 PM 479498 strings_output.txt
FLARE-VM 10/08/2024 15:04:49
PS C:\Users\analysis\Desktop >
15°C Nublado
ENG
LAA
3:04 PM
10/8/202


MSCompressed
{7FC28940-9D31-11D0
FLOSS STACK STRINGS (8)
Proxy error:
0WB4
1WB4
xzzz
xzzz
xzzz
xzzz
0.0.0.0
FLOSS TIGHT STRINGS (7)
7377
w737
373?3;3
37373?3
j:,4;87
EbPZ
BRix
FLOSS DECODED STRINGS (2)
Assertion failed!
File: ../memory.c
C:\Users\analysis\Desktop
λ

Describa los resultados de la inspección del IAT para este binario. ¿Hay alguna importación?

Tras realizar el análisis de la IAT (Import Address Table), observamos que el ejecutable cuenta con 52 importaciones. Es importante destacar que el programa original "putty.exe" también necesita varias de estas importaciones para su correcto funcionamiento. Por ello, hemos seleccionado algunas de las más relevantes y las presentamos a continuación como ejemplo.



¿Vale la pena destacar?

Nombre de la función

Obtener ventana del escritorio

Descripción

GetDesktopWindow se utiliza para obtener un identificador de la ventana del escritorio que cubre toda la pantalla.

Biblioteca

Usuario32.dll

Ataques asociados

Ayudante

Documentación

https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-getdesktopwindow

Creado: 2021-10-30
Última actualización: 2021-10-30
Créditos: mr.d0x

Nombre de la función

ClaveRegCreateA

Descripción

RegCreateKeyA se utiliza para crear una clave de registro específica. Si la clave ya existe, la función la abre.

Biblioteca

Advapi32.dll

Ataques asociados

Ayudante

Documentación

https://docs.microsoft.com/en-us/windows/win32/api/winreg/nf-winreg-regcreatekeya

Creado: 2021-10-30
Última actualización: 2021-10-30
Créditos: mr.d0x

Nombre de la función

Obtener el ID del proceso actual

Descripción

GetCurrentProcessId se utiliza para recuperar el identificador del proceso que realiza la llamada.

Biblioteca

Kernel32.dll

Ataques asociados

Enumeración

Documentación

https://docs.microsoft.com/en-us/windows/win32/api/processthreadsapi/nf-processthreadsapi-getcurrentprocessid

Creado: 2021-10-30
Última actualización: 2021-10-30
Créditos: mr.d0x

Nombre de la función

EliminarArchivoA

Descripción

DeleteFileA se utiliza para eliminar un archivo existente. El malware utiliza esta función para ocultar sus rastros o manipular una aplicación.

Biblioteca

Kernel32.dll

Ataques asociados

Evasión

Documentación

https://docs.microsoft.com/en-us/windows/win32/api/fileapi/nf-fileapi-deletefilea

Creado: 2021-10-30
Última actualización: 2021-10-30
Créditos: mr.d0x

Nombre de la función

ShellExecuteExA

Descripción

ShellExecuteExA se utiliza para realizar una operación en un archivo específico.

Biblioteca

Shell32.dll

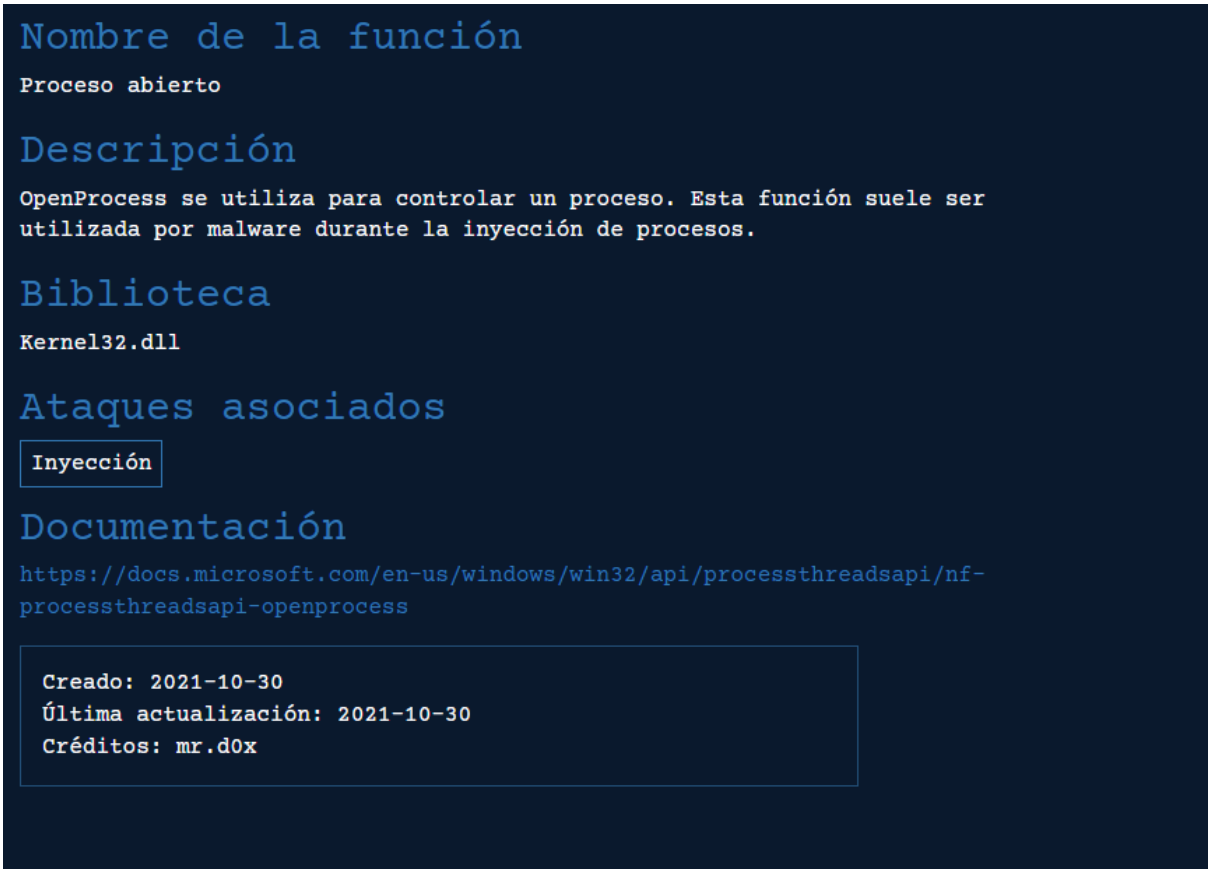
Ataques asociados

Internet

Documentación

https://docs.microsoft.com/en-us/windows/win32/api/shellapi/nf-shellapi-shellexecuteexa

Creado: 2021-10-30
Última actualización: 2021-10-30
Créditos: mr.d0x

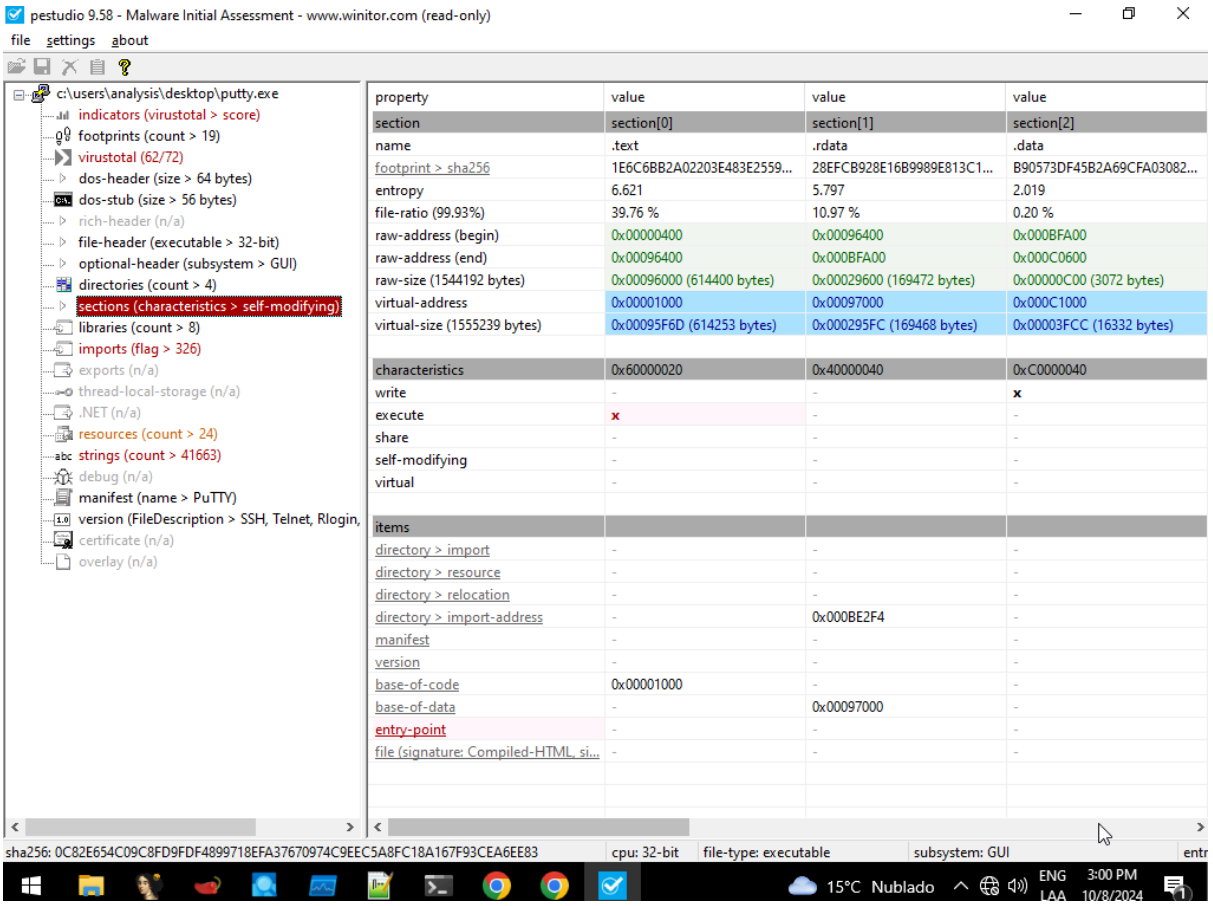


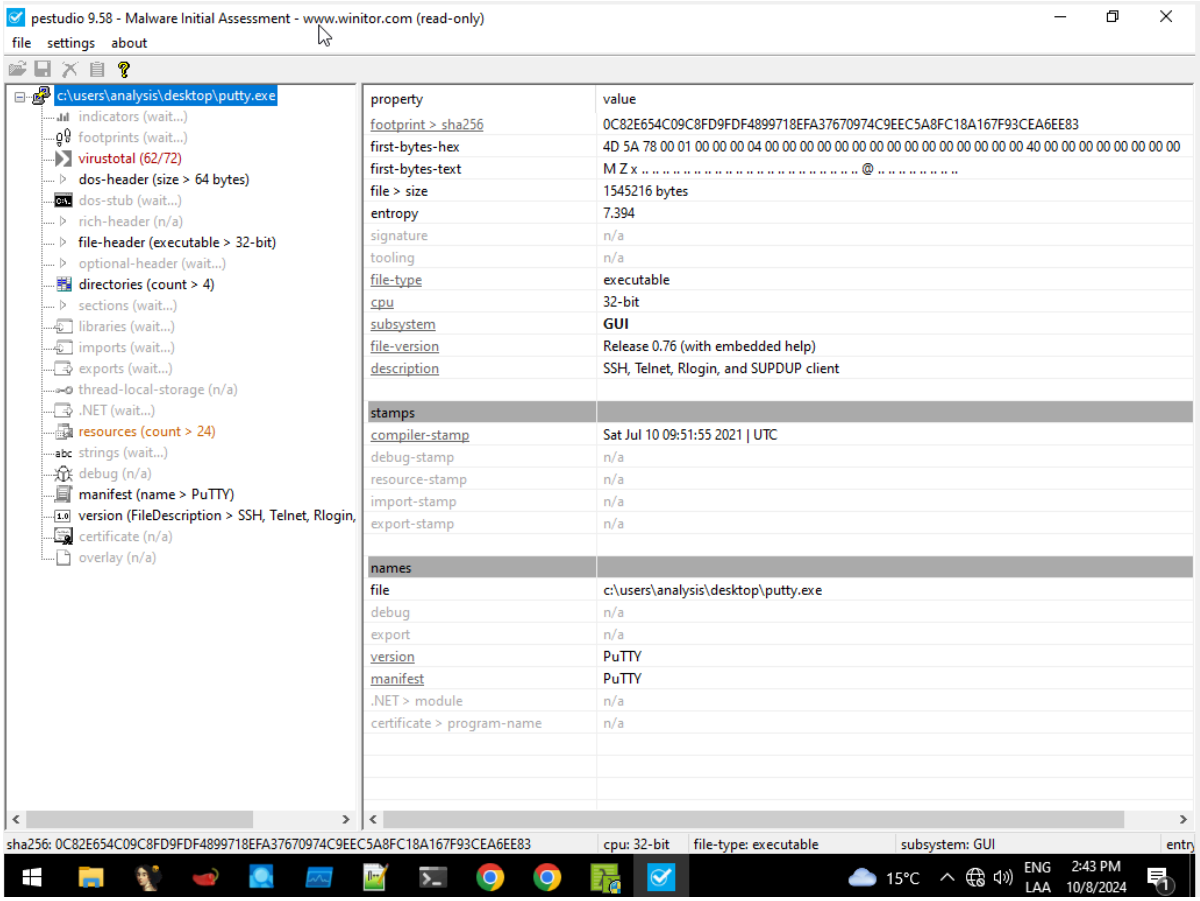
¿Es probable que este binario esté empaquetado?

Es muy probable que el binario no esté empaquetado por dos razones principales:

- Tamaño del archivo: El espacio virtual y el espacio "sólido" que ocupa el programa no muestran una diferencia significativa en cuanto a tamaño. Esto es un indicio claro de que el binario no ha sido sometido a técnicas de empaquetado o compresión, ya que de estar empaquetado, esperaríamos ver una discrepancia considerable entre ambos tamaños.

- Análisis del encabezado IAT: Al revisar el encabezado proporcionado por el análisis de la IAT, no se observan banderas o indicadores que sugieran el uso de empaquetado y desempaquetado. Este tipo de indicadores suelen ser evidentes cuando un binario ha sido modificado de esta manera, y en este caso, la ausencia de dichos elementos refuerza la conclusión de que el programa probablemente no esté empaquetado.


pestudio 9.58 - Malware Initial Assessment - www.winitor.com (read-only)
file settings about
c:\users\analysis\desktop\putty.exe
indicators (virustotal > score)
footprints (count > 19)
virustotal (62/72)
dos-header (size > 64 bytes)
dos-stub (size > 56 bytes)
rich-header (n/a)
file-header (executable > 32-bit)
optional-header (subsystem > GUI)
directories (count > 4)
sections (characteristics > self-modifying)
libraries (count > 8)
imports (flag > 326)
exports (n/a)
thread-local-storage (n/a)
.NET (n/a)
resources (count > 24)
strings (count > 41663)
debug (n/a)
manifest (name > PuTTY)
version (FileDescription > SSH, Telnet, Rlogin,
certificate (n/a)
overlay (n/a)
property value value value
section section[0] section[1] section[2]
name .text .rdata .data
footprint > sha256 1E6C6BB2A02203E483E2559... 28EFCB928E16B9989E813C1... B90573DF45B2A69CFA03082...
entropy 6.621 5.797 2.019
file-ratio (99.93%) 39.76 % 10.97 % 0.20 %
raw-address (begin) 0x00000400 0x00096400 0x000BFA00
raw-address (end) 0x00096400 0x000BFA00 0x000C0600
raw-size (1544192 bytes) 0x00096000 (614400 bytes) 0x00029600 (169472 bytes) 0x00000C00 (3072 bytes)
virtual-address 0x00001000 0x00097000 0x000C1000
virtual-size (1555239 bytes) 0x00095F6D (614253 bytes) 0x000295FC (169468 bytes) 0x00003FCC (16332 bytes)
characteristics 0x60000020 0x40000040 0xC0000040
write - - x
execute x - -
share - - -
self-modifying - - -
virtual - - -
items
directory > import - - -
directory > resource - - -
directory > relocation - - -
directory > import-address - 0x000BE2F4 -
manifest - - -
version - - -
base-of-code 0x00001000 - -
base-of-data - 0x00097000 -
entry-point - - -
file (signature: Compiled-HTML, si... - - -
sha256: 0C82E654C09C8FD9FDF4899718EFA37670974C9EEC5A8FC18A167F93CEA6EE83
cpu: 32-bit
file-type: executable
subsystem: GUI
15°C Nublado
ENG LAA
3:00 PM 10/8/2024


pestudio 9.58 - Malware Initial Assessment - www.winitor.com (read-only)
file settings about
c:\users\analysis\desktop\putty.exe
indicators (wait...)
footprints (wait...)
virustotal (62/72)
dos-header (size > 64 bytes)
dos-stub (wait...)
rich-header (n/a)
file-header (executable > 32-bit)
optional-header (wait...)
directories (count > 4)
sections (wait...)
libraries (wait...)
imports (wait...)
exports (wait...)
thread-local-storage (n/a)
.NET (wait...)
resources (count > 24)
strings (wait...)
debug (n/a)
manifest (name > PuTTY)
version (FileDescription > SSH, Telnet, Rlogin,
certificate (n/a)
overlay (n/a)
property value
footprint > sha256 0C82E654C09C8FD9FDF4899718EFA37670974C9EEC5A8FC18A167F93CEA6EE83
first-bytes-hex 4D 5A 78 00 01 00 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00
first-bytes-text M Z x @
file > size 1545216 bytes
entropy 7.394
signature n/a
tooling n/a
file-type executable
cpu 32-bit
subsystem GUI
file-version Release 0.76 (with embedded help)
description SSH, Telnet, Rlogin, and SUPDUP client
stamps
compiler-stamp Sat Jul 10 09:51:55 2021 | UTC
debug-stamp n/a
resource-stamp n/a
import-stamp n/a
export-stamp n/a
names
file c:\users\analysis\desktop\putty.exe
debug n/a
export n/a
version PuTTY
manifest PuTTY
.NET > module n/a
certificate > program-name n/a
sha256: 0C82E654C09C8FD9FDF4899718EFA37670974C9EEC5A8FC18A167F93CEA6EE83
cpu: 32-bit
file-type: executable
subsystem: GUI
15°C
ENG LAA
2:43 PM 10/8/2024